

**Henri Aho**

# **RASPBERRY PI TIEDONKERUULAITTEENA**

**Case: Nab Labs Oy**

**Opinnäytetyö  
CENTRIA-AMMATTIKORKEAKOULU  
Mediatekniikka  
Marraskuu 2016**

## TIIVISTELMÄ OPINNÄYTETYÖSTÄ

<b>Yksikkö</b> Ylivieska	<b>Aika</b> Marraskuu 2016	<b>Tekijä/tekijät</b> Henri Aho
<b>Koulutusohjelma</b> Mediatekniikka		
<b>Työn nimi</b> TIEDONSIIRTO RASPBERRY PI:LLÄ. Case: Nab Labs Oy		
<b>Työn ohjaaja</b> Hannu Puomio		<b>Sivumäärä</b> 16
<b>Työelämäohjaaja</b> Kaarlo Kärkkäinen		
<p>Opinnäytetyön aiheena oli Raspberry Pi-alustalle suunnitellun etäluettavan ja -hallittavan mittausjärjestelmän toteuttaminen Nab Labs Oy:n ilmatutkimusyksikölle. Työn tavoitteena oli automatisoida mitaustulosten keräämistä ja käsittelyä.</p> <p>Rinnakkain kulkeva teorian ja käytännön kuvaus jakaantuu kolmeen eri osa-alueeseen: Raspberry Pi ja siihen liitetyt laitteet, järjestelmän toiminta ja sen ohjelmointi sekä internet-käyttöliityntä ja www-ohjelmointi.</p> <p>Työn tuloksena syntyi mittalaitteisiin liitettävä järjestelmä, joka toimii pääosin automaattisesti ja on etähallittavissa. Järjestelmään sisältyy internet-sivusto, jonka kautta voidaan muokata mittauksen asetuksia, ladata mitaustuloksia tiedostoon sekä esittää niitä viivakaavioina.</p>		

<b>Asiasanat</b> IoT, PHP, Python, Raspberry Pi, teollinen internet
--

## ABSTRACT

<b>CENTRIA UNIVERSITY OF APPLIED SCIENCES</b> Ylivieska	<b>Date</b> November 2016	<b>Author</b> Henri Aho
<b>Degree programme</b> Media technology		
<b>Name of thesis</b> RASPBERRY PI AS A DATA COLLECTION DEVICE. Case: Nab Labs Oy		
<b>Instructor</b> Hannu Puomio		<b>Pages</b> 16
<b>Supervisor</b> Kaarlo Kärkkäinen		
<p>This thesis covers the building of a remotely readable and manageable measurement-reading system for the air research unit of Nab Labs Oy. The goal was to automate the phases of data collection and processing.</p> <p>The theoretical and practical descriptions, which go hand in hand in this thesis, are divided into three chapters: Raspberry Pi and the devices connected to it, description of the system's operation and programming it, and finally the Internet user interface and www-programming.</p> <p>The result was a system that, once connected to measuring devices, operates mostly automatically and can be remotely managed. The system includes a website, through which one is able to configure the settings of ongoing measurements, download processed data into files and present the results in line diagrams.</p>		

### Key words

Industrial internet, internet of things, PHP, Python, Raspberry Pi

## KÄSITTEIDEN MÄÄRITTELY

<b>A/D-muunnin</b>	Laite, joka muuntaa sille syötetyn analogisen viestin digitaalseksi luvuksi.
<b>GPIO</b>	(General Purpose I/O) Mikroprosessoreissa ja mikrokontrollereissa käytetty portti, joka voidaan ohjelmoida lähettämään tai vastaanottamaan signaalia.
<b>I<sup>2</sup>C</b>	(Inter-Integrated Circuit) Kaksisuuntainen väylä, jota voidaan käyttää ohjaukseen tai tiedonsiirtoon.
<b>NAT</b>	(Network Address Translation) Osoitteenmuunnostekniikka, joka jakaa päätelaitteille ei-julkisen ip-osoitteen.
<b>ppm</b>	(parts per million) Suhteellinen yksikkö, jolla ilmaistaan miljoonasosia jostakin.
<b>SSH</b>	(Secure Shell) Salatun tietoliikenteen protokolla.

**TIIVISTELMÄ**  
**ABSTRACT**  
**KÄSITTEIDEN MÄÄRITTELY**  
**SISÄLLYS**

<b>1 JOHDANTO .....</b>	<b>1</b>
<b>2 RASPBERRY PI ALUSTANA .....</b>	<b>2</b>
2.1 USB-modeemin käyttöönotto .....	3
2.2 Analogia-digitaalimuunnin Pi Plus.....	3
2.3 Paikalliset sensorit.....	4
2.3.1 Lämpötila- ja ilmanpainesensori BMP180 .....	5
2.3.2 Ilmankosteus- ja lämpötilasensori DHT22 .....	5
<b>3 JÄRJESTELMÄN TOIMINTA .....</b>	<b>7</b>
3.1 Python-ohjelmointikieli .....	7
3.2 Tiedon kerääminen ja käsittely.....	7
3.3 Toiminnan automatisointi .....	9
3.4 Tekstiviesti-ilmoitukset.....	10
3.5 Järjestelmän etähallinta .....	10
<b>4 WWW-KÄYTTÖLIITYNTÄ .....</b>	<b>12</b>
4.1 PHP-ohjelmointikieli.....	12
4.2 Mittauksen asetukset .....	13
4.3 Tietojen esittäminen ja tallentaminen .....	13
<b>5 TULOKSET JA POHDINTA .....</b>	<b>15</b>
<b>LÄHTEET .....</b>	<b>16</b>
<b>KUVAT</b>	
KUVA 1. Raspberry Pi 3 Model B .....	2
KUVA 2. ADC Pi Plus liitettynä Raspberry Pi:n päälle .....	4
KUVA 3. Adafruit-piirillisen BMP180-sensorin lukeminen .....	5
KUVA 4. Aosong DHT22-sensorin lukeminen .....	6
KUVA 5. Lämpötila- ja ilmanpainesensoria lukevan Python-ohjelman suorittaminen .....	7
KUVA 6. Python-ohjelma, joka lukee A/D muuntimelta tiedot ja sijoittaa ne muuttujiin .....	8
KUVA 7. Python-ohjelma, joka lähettää A/D-muuntimelta luetut jännitetiedot tietokantaan .....	9
KUVA 8. Esimerkki crontab-tiedostosta .....	10
KUVA 9. Käänteisen SSH-tunnelin luominen ja etähallinta .....	11
KUVA 10. Esimerkki PHP-tiedostosta, johon on liitetty JavaScriptiä .....	12
KUVA 11. Lämpötilaa ja ilmankosteutta esittävä viivakaavio .....	14
KUVA 12. Osa internet-käyttöliitännästä .....	14

## 1 JOHDANTO

Nab Labs Oy on suomalainen mittaus-, analyysi- ja asiantuntijapalveluita tuottava yritys. Opinnäytetyön tilaajana oli Nab Labsin ilmatutkimusyksikkö, jonka tehtäviin lukeutuvat muun muassa teollisuuden päästömittaukset ja ilmanlaadunmittaukset.

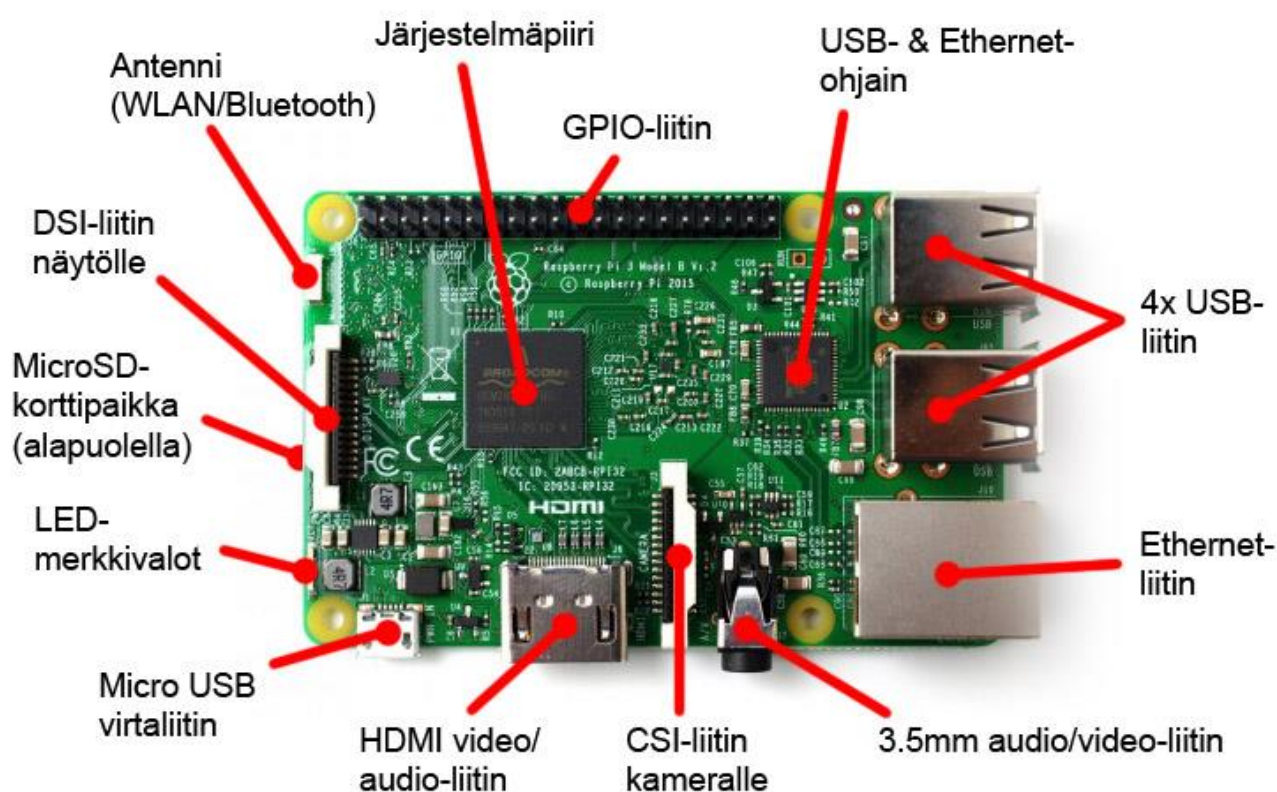
Työn tavoitteena oli uudistaa ja osin automatisoida tapa, jolla päästömittauslaitteilta saatava tieto tallennetaan ja käsitellään jatkokäyttöä varten. Keskeisessä osassa työn suunnittelua olivat mittauksen etähalittavuus sekä kerätyn tiedon automaattinen käsittely, johon aikaisemmin kului paljon työtunteja.

Alussa esitellään alustana käytetty Raspberry Pi sekä siihen liitetyt laitteet ja niiden käyttäminen. Seuraavassa pääluvussa perehdytään Raspberry Pi:n toimintaan järjestelmän keskeisimpänä osana, sekä käydään läpi työkaluja jotka sen mahdollistivat. Viimeisessä käytännön osiossa käsitellään toiminnalliselta osaltaan pääasiassa PHP-ohjelmointikielellä toteutettua internet-käyttöliityntää, sekä tämän kautta tapahtuvaa mittaustulosten tallentamista ja esitystä. Lopuksi pohditaan vielä työn tuloksia sekä matkalla kohdattuja ongelmia ja onnistumisia.

Opinnäytetyön lähteinä toimivat internet-sivustot, joista ei voi nostaa yhtä ylitse muiden. Käytännön toteutuksessa apua löytyi muun muassa Stack Overflow:n ja Raspberry Pi Foundationin keskustelupalstoilta, joilta löytyy paljon neuvoja esimerkiksi ohjelmointiin. Suureksi avuksi oli opetuksella ja ohjauksellaan myös opinnäytetyön ohjaaja, lehtori Hannu Puomio.

## 2 RASPBERRY PI ALUSTANA

Raspberry Pi on pienikokoinen, pääasiassa Linux-käyttöjärjestelmille suunniteltu yhden piirilevyn tietokone, josta on julkaistu useita versioita. Raspberry Pi voidaan kytkeä näytön, näppäimistön ja hiiren kanssa toimimaan normaalin tietokoneen tapaan, ja vaikka se on perinteistä tietokonetta hitaampi, löytyvät siitä kaikki Linux-tietokoneen ominaisuudet. Raspberry Pi:lle on saatavilla myös paljon erilaisia lisälaitteita, kuten kameroita, näyttöjä ja sensoreita. Raspberry Pi suunniteltiin alun perin edulliseksi laitteeksi, joka edistäisi etenkin nuorten ohjelmointi- sekä laiteosaamista. Edullinen hinta, pieni koko sekä hyvä suorituskyky ovat johtaneet suureen suosioon niin elektroniikkaharrastajien kuin ammattilaistenkin keskuudessa. Raspberry Pi:n ympärille on muodostunut vahva yhteisö. Internetistä löytyykin aiheeseen liittyen paljon hyvin dokumentoituja projekteja ja oppaita. (Raspberry Pi Foundation b; Open-source.com.)



KUVA 1. Raspberry Pi 3 Model B (mukaillen Raspberry Pi Foundation a 2016)

Työssä käytettiin Raspberry Pi 2 Model B:tä prototyypin tekemiseen ja myöhemmin Raspberry Pi 3 Model B:tä (KUVA 1) lopullisiin versioihin. Uutta Raspberry Pi 3 Model B:ssä aikaisempaan versioon nähden on tehokkaampi prosessori sekä sisäänrakennettu WLAN- ja Bluetooth-tuki.

## **2.1 USB-modeemin käyttöönotto**

Järjestelmän käyttö vaatii internet-yhteyden tiedon lähettämiseen sekä järjestelmän etähallintaan. Käytötarkoituksesta johtuen kiinteä yhteys ei ollut mahdollinen, joten työssä käytettiin mobiililaajakaistaa ja Huaweiin E3372-USB-modeemia.

USB-modeemeihin on usein tallennettu Windows-käyttöjärjestelmän ajurit. Kun modeemi kytketään PC-tietokoneeseen, se tunnistetaan aluksi USB-muistilaitteena ja ajurit asentuvat automaattisesti, jonka jälkeen laite tunnistetaan modeemina. Linux-käyttöjärjestelmällä tämä ei kuitenkaan toimi, vaan modeemin tila joudutaan vaihtamaan lähettämällä sille komento. (Draisberghof.)

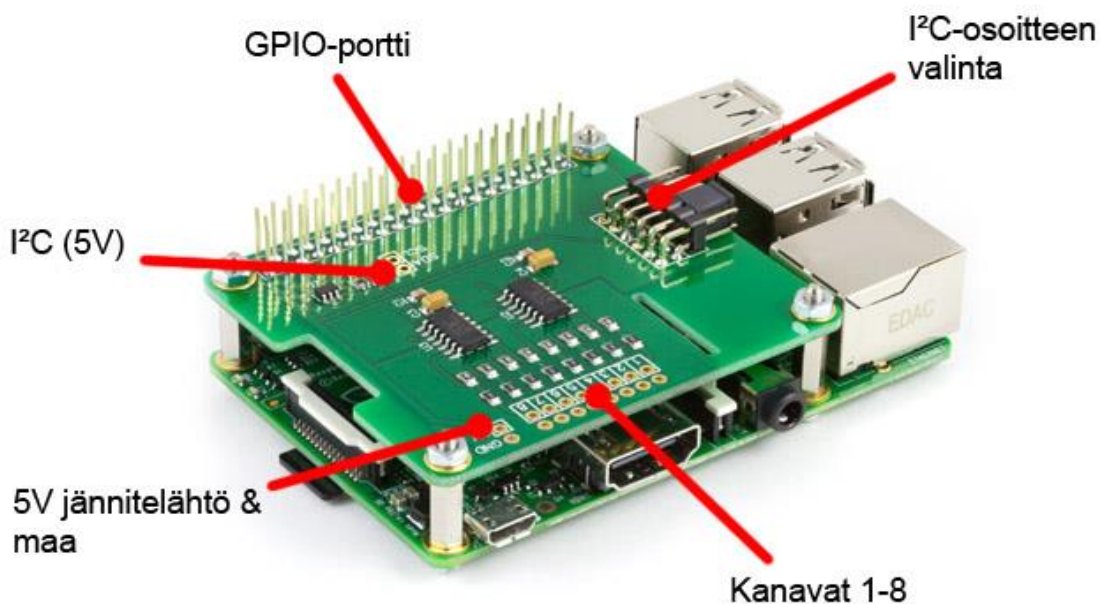
Raspberry Pi osasi muodostaa internet-yhteyden automaattisesti, kun USB-modeemi kytkettiin laitteeseen, mutta sille ei pystynyt lähettämään komentoja komentoriviltä tai ohjelmakoodista. Tämän korjaamiseksi käytettiin USB\_modeswitch-työkalua, jolla laite saatiin vaihdettua Hi-Link-tilasta GSM-modem-tilaan. Tämän jälkeen internet-yhteys pystyttiin muodostamaan käyttämällä Wvdial-komentoriviohjelmaa. Myös tekstiviestien lähettäminen ohjelmakoodista käsin tuli mahdolliseksi.

## **2.2 Analogia-digitaalimuunnin Pi Plus**

Tiedot tulivat mittalaitteilta jänniteviestinä, joten tiedon lukemiseen käytettiin AB Electronics UK:n valmistamaa ADC Pi Plus-korttia. ADC Pi Plus on Raspberry Pi:lle suunniteltu kahdeksankanavainen analogia-digitaalimuunnin eli A/D-muunnin. (SparkFun).

ADC Pi Plus kytketään Raspberry Pi:n GPIO-porttiin, minkä jälkeen tarvitsee vain ottaa I<sup>2</sup>C-väylä käyttöön ja asentaa joitakin työkaluja. AB Electronics UK:lta löytyy valmis Python-kirjasto, jonka avulla tiedon lukeminen on helppoa (KUVA 6).





KUVA 2. ADC Pi Plus liitettynä Raspberry Pi:n päälle (mukaillen AB Electronics UK)

Kuvaan 2 on merkitty GPIO-porttia jatkavat pinnit, joiden ansiosta ADC Pi Plussan päälle voidaan liittää muita moduuleja. ADC Pi Plus on suunniteltu siten, että niitä voidaan käyttää useampaa yhtäaikaaisesti, tai muiden I<sup>2</sup>C:tä käyttävien laitteiden kanssa. Kuvassa 2 näkyy myös I<sup>2</sup>C-osoitteen valitsin, jolla laitteen I<sup>2</sup>C-osoite voidaan vaihtaa kytkemällä jumpperit eri pinnien välille.

## 2.3 Paikalliset sensorit

Varsinaisen mittalaitteiston lisäksi työhön liitettiin myös kaksi suoraan Raspberry Pi:hin kytkettyä sensoria, joilla mitattiin paikallista lämpötilaa, ilmakeuhetta ja -painetta. Sensorit liitettiin ADC Pi Plus-kortin päälle Raspberry Pi:n GPIO-porttiin. Molemmille sensoreille löytyy Adafruitin internet-sivuston kautta valmiit Python-kirjastot, joiden avulla tiedot saadaan luettua ohjelmakoodissa.

### 2.3.1 Lämpötila- ja ilmanpainesensori BMP180

BMP180 on Boschin valmistama sensori, joka mittaa lämpötilaa ja ilmanpainetta. Sensorilta saadaan myös laskettuna arvona korkeus merenpinnasta sekä merenpinnan tasolle korjattu ilmanpaine. Korkeuslukemasta huomattiin, että se ei ole kovinkaan tarkka, mutta sitä ei tässä työssä tarvittu.

Työssä käytettiin Adafruitin valmistamaa piirilevyä, johon tämä sensori on liitetty, jolloin sitä voidaan lukea Raspberry Pi:n GPIO-portin kautta (KUVA 3). BMP180 käyttää samoja GPIO-portin I<sup>2</sup>C-pinnejä kuin ADC Pi Plus, mutta koska I<sup>2</sup>C-väylässä laitteilla on eri osoitteet, niitä voidaan käyttää yhtäaikaista. BMP180:ltä saatiin referenssimittauksissa yhteneväisempiä lämpötilalukemia kuin DHT22:lta, joten lämpötilalukema otettiin tältä sensorilta.

```
import Adafruit_BMP.BMP085 as BMP085
sensor = BMP085.BMP085()

temp = sensor.read_temperature()
pressure = sensor.read_pressure()
altitude = sensor.read_altitude()
sealevel_pressure = sensor.read_sealevel_pressure()
```

KUVA 3. Adafruit-piirillisen BMP180-sensorin lukeminen

### 2.3.2 Ilmankosteus- ja lämpötilasensori DHT22

DHT22 on Aosong Electronicsin valmistama ilmankosteus- ja lämpötilasensori, joka BMP180:stä poiketen käyttää omaa datapinniä GPIO-portissa. DHT22 vaatii ylösvetovastuksen data- ja virtapinnien välille. Tähän olisi voitu ohjelmallisesti hyödyntää Raspberry Pi:n sisäänrakennettuja vastuksia, mutta vaikutti yksinkertaisemmalta juottaa pinnien välille erillinen 4,7 Kilo-ohmin vastus. Tältä sensorilta käytettiin ainoastaan ilmankosteuslukemaa.

```
import Adafruit_DHT
sensor = Adafruit_DHT.DHT22
pin = 22
humidity, temperature = Adafruit_DHT.read_retry(sensor, pin)
```

KUVA 4. Aosong DHT22-sensorin lukeminen

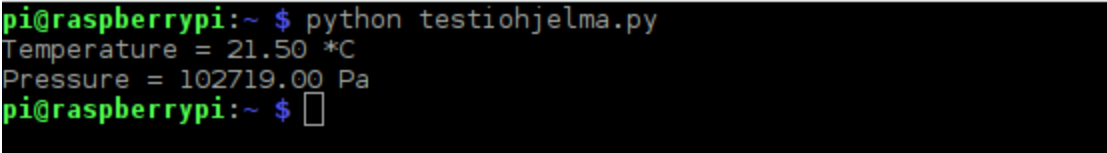
Kuvasta 4 nähdään, että ohjelmakoodissa täytyy määritellä, mihin GPIO-portin pinniin DHT22-sensorin datapinni on liitetty. DHT22:lta molemmat luettavat tiedot saadaan yhdellä ohjelmalauseella.

### 3 JÄRJESTELMÄN TOIMINTA

Tavoitteena oli luoda järjestelmä, joka ei perustoiminnan osalta vaadi käyttäjältä muuta kuin virtajohdon kiinnittämisen. Toiminta perustuu pääasiassa Python-ohjelmointikielellä kirjoitettuihin skripteihin, joita suoritetaan automaattisesti järjestelmän käynnistyessä ja sen jälkeen asetetuin väliajoin. Tapauskohtaiset asetukset säädetään internet-sivuston ja tietokantojen kautta, jolloin itse laitteeseen ei tarvitse koskea.

#### 3.1 Python-ohjelmointikieli

Python on tehokas, korkean tason tietorakenteita käyttävä olio-ohjelmointikieli, jonka kehitti Guido van Rossum vuonna 1989 (Programiz). Python on tulkattava ohjelmointikieli, eli se on valmis suoritettavaksi heti kirjoittamisen jälkeen, ilman erillistä kääntämistä (KUVA 5). Monet ohjelmoijat suosivatkin Pythonia, koska se mahdollistaa ohjelmakoodin nopean testaus- ja korjausvaiheen, ja näin parantaa tuottavuutta. Pythonia pidetään myös helppona kielenä oppia yksinkertaisine syntakseineen. (Python Software Foundation.)



```
pi@raspberrypi:~ $ python testiohjelma.py
Temperature = 21.50 *C
Pressure = 102719.00 Pa
pi@raspberrypi:~ $
```

KUVA 5. Lämpötila- ja ilmanpainesensoria lukevan Python-ohjelman suorittaminen

#### 3.2 Tiedon kerääminen ja käsittely

Mittaustulokset luetaan A/D-muuntimelta ja sensoreilta valmiiden Python-kirjastojen avulla. Kirjastolla tarkoitetaan ohjelmoinnissa valmiiksi kirjoitettua koodia, joka voidaan liittää osaksi ohjelmaa, jolloin sen sisältämiä toimintoja ei tarvitse ohjelmoida alusta alkaen itse (Techopedia). Pythonissa kirjastot tuodaan ohjelmaan import-lauseella, jonka jälkeen sen toimintoja voidaan käyttää. Kuvassa 6 nähdään esimerkiksi A/D-muuntimen valmistajan tarjoaman Python-kirjaston sisältämän ”read\_voltage”-funktion käyttö jännitetietojen lukemiseen kanavilta.

```
import sys
sys.path.append('/home/pi/ABElectronics_Python_Libraries/ADCPi/')
from ABE_ADCPi import ADCPi
from ABE_helpers import ABEHelpers
import MySQLdb

i2c_helper = ABEHelpers()
bus = i2c_helper.get_smbus()
adc = ADCPi (bus, 0x68, 0x69, 18)

ch1 = adc.read_voltage(1)
ch2 = adc.read_voltage(2)
ch3 = adc.read_voltage(3)
ch4 = adc.read_voltage(4)
ch5 = adc.read_voltage(5)
ch6 = adc.read_voltage(6)
ch7 = adc.read_voltage(7)
ch8 = adc.read_voltage(8)
```

KUVA 6. Python-ohjelma, joka lukee A/D muuntimelta tiedot ja sijoittaa ne muuttujiin

Python-ohjelma muuntaa luetut jännitetiedot ppm-arvoiksi laskukaavalla, johon haetaan asetetut laskenta-arvot tietokannasta MySQL-kyselyllä. Laskennan jälkeen lasketut arvot, raakatiedot, laskentaan käytetyt arvot ja aikaleima lähetetään tietokantaan palvelimelle.

```

import MySQLdb

db = MySQLdb.connect(
    host="esimerkkipalvelin.fi",
    user="testUser",
    passwd="testPw",
    db="testDB",
    port=3306)
curs = db.cursor()

sql = str(("INSERT INTO testitaulu (aika,ch1,ch2,ch3,ch4,ch5,ch6,ch7,ch8)
VALUES (NOW(), '{0}', '{1}', '{2}', '{3}', '{4}', '{5}', '{6}', '{7}')"
).format(ch1,ch2,ch3,ch4,ch5,ch6,ch7,ch8)

try:
    curs.execute(sql)
    db.commit()

except MySQLdb.Error, e:
    print e
    print "Error %d:" % (e.args[0], e.args[1])
    db.rollback()
    sys.exit(1)

curs.close()
db.close()

```

KUVA 7. Python-ohjelma, joka lähettää A/D-muuntimelta luetut jännitetiedot tietokantaan

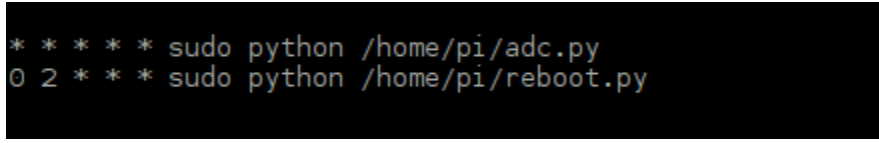
### 3.3 Toiminnan automatisointi

Toiminnan automatisointi oli järjestelmän suunnittelussa keskeisin osa-alue, sillä sen koko toiminta perustuu jonkinasteiseen automaatioon. Käyttäjän tehtäväksi pyrittiin jättämään vain tapauskohtaisten arvojen päivittäminen verkkolomakkeiden kautta, sekä valmiiksi käsiteltyjen tietojen tallentaminen jatkokäyttöä varten.

Järjestelmää suunnitellessa piti varautua myös siihen, että internet-yhteys voi välillä katkeilla, joten ohjelmakoodiin sisällytettiin internet-yhteyden tarkistus tietojen lukemisen jälkeen. Mikäli yhteys ei toimi, yritetään ensin uudelleenyhdistämistä. Jos se ei onnistu, tiedot kirjoitetaan tiedostoon, josta ne luetaan ja lähetetään, kun yhteys taas toimii. Tällöin mittaustulokset saadaan talteen myös pidempiaikaisen katkoksenkin ajalta.

Ohjelmien ajastukseen käytettiin Cronia. Cron on Unix-tyyppisistä käyttöjärjestelmistä löytyvä taustaprosessi, joka mahdollistaa tehtävien suorittamisen asetetuin väliajoin. Crontab on tiedosto, johon tehtävät ja niiden ajoitukset syötetään. Cronin avulla voidaan esimerkiksi asettaa järjestelmä käynnistymään uudelleen tietyinä ajankohtana. (Admin's Choice.)

Cron oli keskeisessä osassa järjestelmän automatisoinnissa. Järjestelmän käynnistyessä sen kautta avataan internet-yhteys, minkä jälkeen ryhdytään suorittamaan itse mittausta ja mittaustulosten käsittelyä. Cronilla myös tarkistetaan internet-yhteys ja ylläpidetään etähallintayhteyden mahdollisuutta.



```
* * * * * sudo python /home/pi/adc.py
0 2 * * * sudo python /home/pi/reboot.py
```

KUVA 8. Esimerkki crontab-tiedostosta

Kuvassa 8 nähdään esimerkkinä crontab-tiedoston rivit, joista ensimmäinen suorittaa ”adc.py”-ohjelman joka minuutti ja toinen suorittaa ”reboot.py”-ohjelman joka päivä kello 02:00. Rivin viisi ensimmäistä merkkiä tarkoittavat minuutteja, tunteja, kuukauden päivää, kuukautta ja viikonpäivää.

### 3.4 Tekstiviesti-ilmoitukset

Mittaustulosten laskennan jälkeen ohjelma tarkistaa tietokannasta, onko hälytysrajoja otettu käyttöön. Mikäli laskettu arvo alittaa tai ylittää asetetun minimi- tai maksimi-arvon, lähetetään asetettuun puhelinnumeroon USB-modeemin kautta tekstiviesti, joka sisältää hälytyksen aiheuttaneen mittauskohteen sekä sille lasketun arvon. Jotta tekstiviestejä ei lähetetä useita peräkkäin, täytyy hälytys käydä kuittaamassa internet-sivustolta ennen kuin uusia lähetetään.

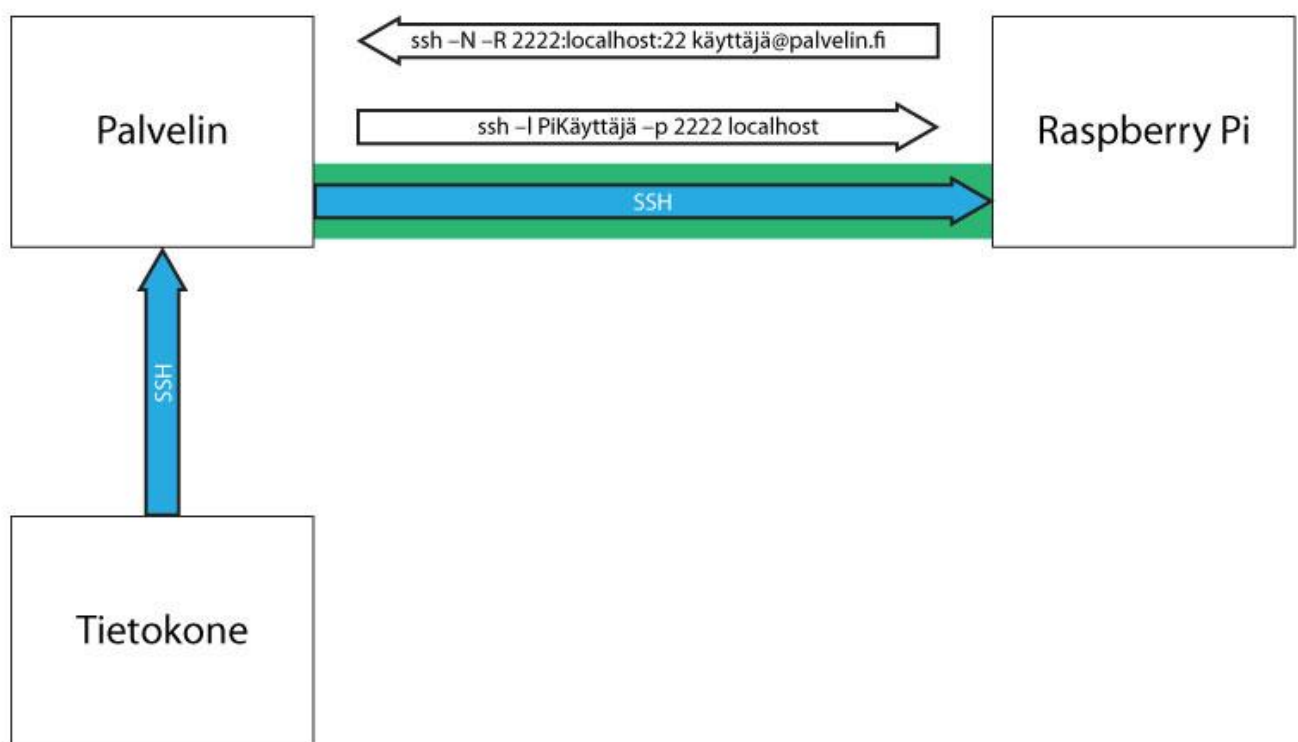
### 3.5 Järjestelmän etähallinta

Mikäli järjestelmä kaipaa korjausta tai parannuksia, SSH-etäyhteydellä päästään esimerkiksi muokkaamaan ohjelmakoodia Raspberry Pi:ssä vaikka meneillään olevan mittauksen aikana. Mobiililaajakaistan

NAT-osoitteenmuunnos-tekniikasta johtuen Raspberry Pi:hin ei pystytä ottamaan suoraan SSH-yhteyttä. Raspberry Pi:n IP-osoite ei myöskään ole useimmiten tiedossa, joten palvelimen ja Raspberryn Pi:n välille luotiin käänteinen SSH-tunneli.

Kuten kuvassa 9 on esitetty, tunneli luodaan suorittamalla kohdelaitteella (Raspberry Pi) komento, joka käskää palvelimen kuunnella esimerkiksi porttia 2222 ja ohjata sinne tulevan yhteyden takaisin Raspberry Pi:n porttiin 22. Tämän jälkeen etäyhteys voidaan avata joko suoraan palvelimelta, tai joltakin toiselta tietokoneelta luomalla ensin tavallinen SSH-yhteys palvelimeen.

Jotta tunneli voidaan luoda kohdelaitteen ollessa fyysisesti kauempana, on otettava käyttöön avainpohjainen autentikointi salasanan sijasta. Tämä tapahtuu siten, että kohdelaitteella suoritetaan komento, joka luo yksityisen avaimen (id\_rsa) sekä julkisen avaimen (id\_rsa.pub). Yksityinen avain siirretään kohdelaitteen ".ssh"-kansioon ja julkisen avaimen sisältö kopioidaan palvelimen "authorized\_keys"-tiedostoon. Nyt kohdelaite voidaan asettaa ylläpitämään tunnelia automaattisesti, ilman, että sen tarvitsee syöttää salasanaa palvelimelle.



KUVA 9. Käänteisen SSH-tunnelin luominen ja etähallinta



## 4 WWW-KÄYTTÖLIITYNTÄ

Järjestelmä suunniteltiin siten, että kerättyyn tietoon päästään käsiksi internet-sivuston kautta miltä tahansa tietokoneelta. Tiedot saadaan ladattua Excel-tiedostoon ja viivakaavioihin suoraan internet-selaimen kautta.

### 4.1 PHP-ohjelmointikieli

PHP on yleisesti käytetty www-ohjelmointikieli, jota voidaan sekoittaa HTML:n kanssa. Toisin kuin HTML, PHP-koodi suoritetaan palvelimella ennen kuin PHP:tä sisältävä sivu ladataan selaimessa. PHP on skriptikieli, eli sitä ei tarvitse kääntää ennen suorittamista. PHP:llä voidaan esimerkiksi käsitellä tietokantoja sekä lähettää tietoja lomakkeelta toiselle sivulle. (Wired.)

```
<script type="text/javascript">
function popup() {
    window.alert("Tietokanta tyhjennetty!");
    document.location.href = "edellinensivu.php";
}
</script>
<?php
include('yhteys.php'); // yhteys tietokantaan erillisellä php-tiedostolla

$taulu = $_POST['selectTaulu'];
$tyhjenna = "TRUNCATE $taulu";

if (mysqli_query($yhdistä,$tyhjenna)) {
    echo '<script type="text/javascript">'
        , 'popup();'
        , '</script>';
} else {
    echo "Error: " . $tyhjenna . "<br>" . $yhdistä->error;
}
$yhdistä->close();
?>
```

KUVA 10. Esimerkki PHP-tiedostosta, johon on liitetty JavaScriptiä

Kuvassa 10 on esitetty yksinkertaistettuna työssä käytetty PHP-skripti, joka tyhjentää valitun taulun tietokannasta. Lomakesivulta lähetetään haluttu taulu tälle PHP-skriptille POST-metodilla, minkä jälkeen se sijoitetaan muuttujana MySQL-kyselyyn. Jos tyhjentäminen onnistuu, näytetään käyttäjälle JavaScriptillä toteutettu ponnahdusikkuna ja palataan edelliselle sivulle.

## 4.2 Mittauksen asetukset

Sivuston kautta pystytään muokkaamaan arvoja, joiden perusteella Raspberry Pi:ssä suoritettava ohjelmakoodi laskee mitatut raakatiedot ppm-arvoiksi. Sivustolla voidaan myös laittaa jokaiselle mittalaitteelle merkintä, jos laitteella on menossa esimerkiksi säätövaihe, jolloin mittaustulokset poikkeavat normaalista.

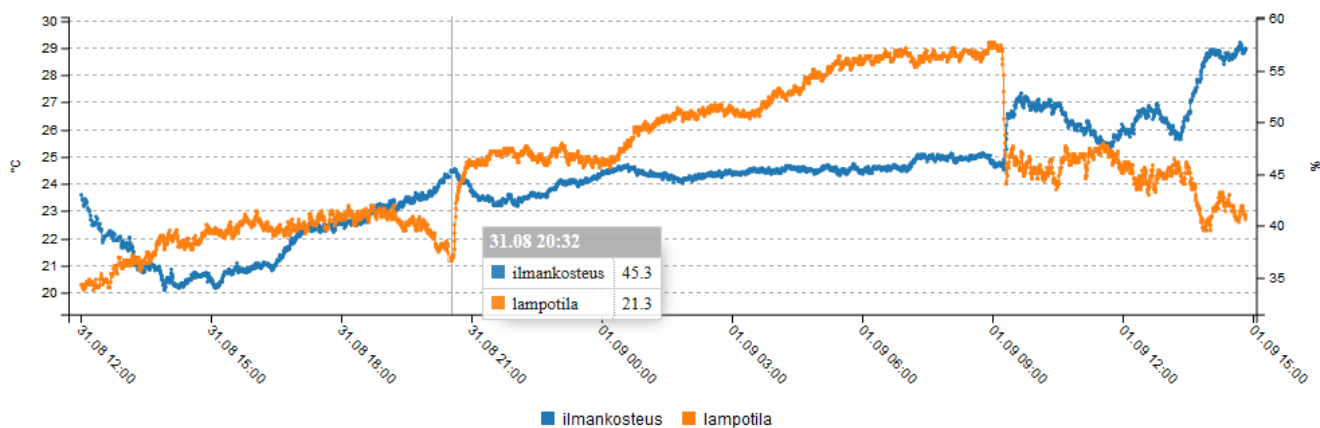
Asetuksista (KUVA 12) saadaan asetettua jokaiselle laitteelle puhelinnumero, johon lähetetään haluttaessa tekstiviesti-ilmoituksia. Hälytykset voidaan kytkeä päälle jokaiselle laitteelle ja mitattavalle asialle erikseen, sekä asettaa arvolle joko minimi tai maksimi, tai molemmat.

## 4.3 Tietojen esittäminen ja tallentaminen

Internet-sivustolla oli tarkoitus pystyä esittämään mittaustuloksia siten, että niitä voidaan esitellä asiakkaille sekä seurata, eteneekö mittaus oikein. Tiedot piti saada myös tallennettua Excel-tiedostoon jatkokäsittelyä ja arkistointia varten.

Tietojen esittäminen toteutettiin viivakaavioina (KUVA 10) sekä taulukkona käyttäen JavaScriptiä, PHP:tä ja HTML:ää. Viivakaaviot toteutettiin C3.js-JavaScript kirjastoa käyttäen. Eri mittausten lukumäärästä ja eroavista asteikoista johtuen tiedot jaettiin kolmeen eri viivakaavioon, jotta tulkitseminen olisi helpompaa. Tietojen tallentaminen Excel-tiedostoon onnistui PHPExcel-PHP -kirjaston avulla.

Tietojen hakeminen kaavioille ja taulukoihin tapahtuu MySQLi-kyselyillä PHP:n kautta. Hauille voidaan myös asettaa aikaväli DatePicker-JavaScript -kirjaston avulla luodulla kalenterilomakkeella, sekä valita, minkä Raspberry Pi:n tiedot haetaan.



KUVA 11. Lämpötilaa ja ilmankosteutta esittävä viivakaavio

Kuvassa 11 esitetyn kaavion ominaisuuksia ovat muun muassa kuvassa näkyvät tarkemmat tiedot kurssin kohdistuksella sekä zoomaus hiiren rullalla. Useita kuvaajia sisältävässä kaaviossa kuvaajia voidaan myös piilottaa tai korostaa, jos halutaan tutkia tarkemmin jotakin asiaa.

### Mittausasetukset:

Katsele/muokkaa säätöarvoja laitteille:

### Hälytykset:

Laite	Kuittaus
Raspberry 1	
Raspberry 2	
Raspberry 3	<input type="button" value="Kuittaa"/>

### SMS-asetukset:

Laite	Nimi	Numero
Raspberry Pi	Henri Aho	
Raspberry Pi 2	Toinen numero	
Raspberry Pi 3	Kolmas numero	
Raspberry Pi 4	ASD	5555555555

### Vaihda numero:

KUVA 12. Osa internet-käyttöliitännästä

## 5 TULOKSET JA POHDINTA

Työn tuloksena syntyi pitkälti alkuperäisen suunnitelman mukainen, toimiva järjestelmä. Työn edetessä joitakin asioita tehtiin eri tavalla ja joitakin ominaisuuksia lisättiin, mutta pääperiaate ei juurikaan muuttunut.

Edeltävään järjestelmään verrattuna suurimmat muutokset olivat asetusten etähallittavuus sekä mittauksen jälkeinen tietojenkäsittely, joka hoituu pääasiassa automaattisesti. Internet-sivuston kautta mittauksen kulkua voidaan myös seurata reaaliajassa, jolloin voidaan havaita mahdollisia ongelmia.

Raspberry Pi:llä toteutetun järjestelmän etuja ovat edulliset hankintakustannukset, ja sitä voidaan myös kustomoida hyvin tarpeiden mukaiseksi. Toisaalta siihen liittyy myös haasteita, sillä kaikki järjestelmään käytetyt ohjelmat ja laitteet eivät aina toimineet odotetulla tavalla yhdessä. Kaikkiin kohdattuihin ongelmiin kuitenkin lopulta löytyi ratkaisu.

Työhön jäi vielä myös jatkokehitysmahdollisuuksia. Yksi näistä on paikallisten sensorien tarkkuuden tutkiminen ilmanpaineen ja ilmankosteuden osalta, sekä mahdollisesti parempien vaihtoehtojen testaaminen. Toinen kehityskohde on laitteen kotelointi siten, että se kestää kuljetusta ja karumpiakin olosuhteita mittauksen aikana. Koteloinnin yhteydessä voidaan myös suunnitella helpompikäyttöiset ja kestävämmät liitännät mittalaitteille.

Opinnäytetyön aihe oli todella mielenkiintoinen ja opettavainen. Työtä tehdessä opin paljon lisää paitsi ohjelmoinnista ja Linux-käyttöjärjestelmästä, myös elektroniikasta ja muista asioista. Opinnäytetyö antoi suuntaa siitä, minkälaista työtä haluaisin tehdä tulevaisuudessa. Työn ohessa huomasin lisäksi monia mahdollisia sovelluskohteita Raspberry Pi:lle, kotiautomaatiosta teolliseen internettiin.

## LÄHTEET

AB Electronics UK. Saatavissa: <https://www.abelectronics.co.uk/p/56/ADC-Pi-Plus-Raspberry-Pi-Analogue-to-Digital-converter>. Viitattu 18.10.2016.

Admin's Choice. Crontab – Quick Reference. Saatavissa: <http://www.adminschoice.com/crontab-quick-reference>. Viitattu 20.7.2016.

Draisberghof. USB\_ModeSwitch - Handling Mode-Switching USB Devices on Linux. Saatavissa: [http://www.draisberghof.de/usb\\_modeswitch/](http://www.draisberghof.de/usb_modeswitch/). Viitattu 30.6.2016.

Opensource.com. What is a Raspberry Pi? Saatavissa: <https://opensource.com/resources/what-raspberry-pi>. Viitattu 25.6.2016.

Programiz. Python Programming Tutorial. Saatavissa: <http://www.programiz.com/python-programming>. Viitattu 29.7.2016.

Python Software Foundation. What is Python? Executive Summary. Saatavissa: <https://www.python.org/doc/essays/blurbl/>. Viitattu 29.7.2016.

Raspberry Pi Foundation a. Saatavissa: <https://www.raspberrypi.org/wp-content/uploads/2016/03/pi3-500x333.jpg>. Viitattu 6.10.2016.

Raspberry Pi Foundation b. What is a Raspberry Pi? Saatavissa: <https://www.raspberrypi.org/help/what-is-a-raspberry-pi/>. Viitattu 25.6.2016.

SparkFun. Analog to Digital Conversion. Saatavissa: <https://learn.sparkfun.com/tutorials/analog-to-digital-conversion>. Viitattu 26.6.2016.

Techopedia. Software Library. Saatavissa: <https://www.techopedia.com/definition/3828/software-library>. Viitattu 11.10.2016.

Wired. PHP Tutorial for Beginners. Saatavissa: [http://www.wired.com/2010/02/php\\_tutorial\\_for\\_beginners/](http://www.wired.com/2010/02/php_tutorial_for_beginners/). Viitattu 12.8.2016.